Type-2 fuzzy controller in ZigBee network

MICHAL BLAHO, MARTIN URBAN, MARTIN FOLTIN
Institute of control and industrial informatics
Faculty of Electrical Engineering and Information Technology
Slovak University of Technology
Ilkovičova 3, 812 19 Bratislava
SLOVAK REPUBLIC
michal.blaho@stuba.sk, martin.urban@stuba.sk

Abstract: - Networked control systems (NCS) have been in attention for many researchers for the past ten years. Professional solutions like PROFINET or IWLAN are followed with cheaper and popular networks like ZigBee. Multiple control methodologies have been developed for this kind of systems. In this paper we threatened network disadvantages as uncertainty and we reduced them with type-2 fuzzy logic controller. We compared results with PID controller.

Key-Words: - Networked control systems, ZigBee, Fuzzy type-2 logic, PID controller, Type-2 fuzzy controller, uncertainty, TrueTime

1 Introduction
Networked Control Systems (NCSs) are one type of distributed control systems where sensors, actuators, and controllers are interconnected by communication networks. Many authors interested in topics of NCS studied problems of NCS like control of the variable transport delay and packet dropouts [1,2,3].

The traditional wired networks are most common and reliable in this area. New possibilities for modern data networks (cable or wireless) occurred. Data networks and protocols have been developed specially for networked control systems. Cheaper but very popular networks like ZigBee have been followed solutions for industrial applications like PROFINET or IWLAN [4].

Fuzzy logic has been evolving for more than 45 years now. Many authors studied properties of fuzzy logic systems and applied their results in control. Human decisions can be included into rules and therefore we can control the systems without knowing their mathematical model [5].

Last decade scientists focused on type-2 fuzzy logic. Type-2 fuzzy logic can better model and minimize effect of different kinds of uncertainties. NCS also contain different types of uncertainties like before mentioned transport delays or packet dropouts. Therefor usage of type-2 fuzzy systems in NCS can be beneficial [6].

In this paper we compared performance of PID controller and type-2 fuzzy controller in ZigBee wireless network. For each of this controller we simulated different settings of network.

2 Networked control systems
Computer data networks have a long history. Lots of new communications media and protocols have been developed and improved for many years. This led to many benefits as remote data transfers and data exchanges, reduce wiring, costs of medias and easier maintenance. Considering this benefits data networks became popular in control systems [7].

Multiple sensors and actuators are connected to a centralized controller via shared communication medium in a NCS. Controller, sensors and actuators transmit information and control signal via network as you can see on figure 1 [1].

Fig. 1 NCS block diagram

On the other hand integrating computer networks into control systems brings also disadvantages. Regardless of the type of network used, the networked control system (NCS) performance is always affected by network delays or packet dropouts. These two keys issues are widely known to degrade the performance or robustness of a control system [7].
There are different networks suitable for NCS in industrial applications. Data can be transferred through cable network (e.g. CAN, EtherCAT, PROFINET) or wireless network (e.g. ZigBee, IWLAN). We focused on ZigBee network in this paper.

3 ZigBee
ZigBee is the only standards-based wireless 802.15.4 technology designed to address the unique needs of low-cost, low-power wireless sensor and control networks. ZigBee protocol features include: support for multiple network topologies such as point-to-point, point-to-multipoint and mesh networks, low duty cycle – provides long battery life, low latency, up to 65,000 nodes per network, encryption for secure data connections. ZigBee found usage in many applications like building automation, remote control, health care, home automation and many others [8, 9, 10].

3 Type-2 fuzzy logic
Zadeh has introduced fuzzy sets in 1965. Ten years later he presented concept of type-2 fuzzy sets. Since classical (type-1) sets have been used with success in many fields but not many researchers focused on type-2 fuzzy sets. Mendel returned to the concept of type-2 fuzzy sets and for the last ten years with other researchers have extended this theory [11].

Type-1 fuzzy sets have limited capabilities to directly handle data uncertainties, where handle means to model and minimize the effect of. Type-2 fuzzy sets have grades of membership that are fuzzy, so it could be called a "fuzzy-fuzzy set".

A type-2 fuzzy set in the universal set $X$ is denoted as $\tilde{A}$, which is characterized by a type-2 membership function $\mu_{\tilde{A}}(x)$ in (1). The $\mu_{\tilde{A}}(x)$ can be referred as a secondary membership function (MF) or also referred as secondary set, which is a type-1 set in [0,1]. In (1) $f_x(u)$ is a secondary grade, which is the amplitude of a secondary MF; i.e. $0 \leq f_x(u) \leq 1$. The domain of a secondary MF is called the primary membership of $x$. In (1), $J_x$ is the primary membership of $x$, where $u \in J_x \subseteq [0,1]$ for all $x \in X$; $u$ is a fuzzy set in [0,1], rather than a crisp point in [0,1]. [12]

$$\tilde{A} = \int_{X \subseteq [0,1]} u_{\tilde{A}}(x)/x = \int_{x \subseteq [0,1]} \left[ \int_{u \subseteq [0,1]} f_x(u)/u \right]/x, \ J_x \subseteq [0,1]$$ (1)

When $f_x(u) = 1, \ \forall u \in J_x \subseteq [0,1]$, then the secondary MFs are interval sets such that $\mu_{\tilde{A}}(x)$ in (1) can be called an interval type-2 MF [...]. Therefore, type-2 fuzzy set $\tilde{A}$ can be rewritten as

$$\tilde{A} = \int_{x \subseteq [0,1]} \left[ \int_{u \subseteq [0,1]} 1/u \right]/x, \ J_x \subseteq [0,1]$$ (2)

An example of interval type-2 fuzzy set that is often used (we also used it in our computation) is Gaussian membership function with uncertain mean and fixed standard deviation (3). This function is in the figure 2.

$$\mu_{\tilde{A}}(x) = \exp \left[ -\frac{1}{2} \left( \frac{x - m}{\sigma} \right)^2 \right], \ m \in [m_1, m_2]$$ (3)

As you can see in figure 2 region of the Gaussian MF with uncertain mean can be bounded by upper $\tilde{\mu}_{\tilde{A}}(x)$ and lower $\mu_{\tilde{A}}(x)$ MF and it is called footprint of uncertainty (FOU).

Type-2 fuzzy logic system is similar to type-1 fuzzy logic system and is shown in figure 3. This system consists of these five main parts: fuzzifier, rule base, inference engine, type-reducer, and defuzzifier.

![Fig. 3 Type-2 fuzzy logic system](image)

The type-2 fuzzy rule base consists of a collection of IF-THEN rules. With M rules and the rule of a
The type-2 relation between the input space and the output space can be expressed as:

\[ R^i: \text{IF } x_i \text{ is } F^i_1 \text{ and } \ldots \text{ and } x_P \text{ is } F^i_P, \text{ THEN } y \text{ is } G^i, i = 1, \ldots, M \]  

(4)

The inference engine combines rules and gives a mapping from input type-2 fuzzy sets to output type-2 fuzzy sets. The output of inference engine block is a type-2 set. The firing strength \( f^i \) for the \( i \)th rule can be an interval type-1 set expressed as

\[ f^i = [f^i_1, f^i_2] \]

where

\[ f^i_1 = \mu_{F^i_1}(x_1) \times \mu_{F^i_2}(x_2) \times \ldots \times \mu_{F^i_P}(x_P) \]

(5)

\[ f^i_2 = \mu_{F^i_1}(x_1) \times \mu_{F^i_2}(x_2) \times \ldots \times \mu_{F^i_P}(x_P) \]

(6)

To reduce type-2 fuzzy set to type-1 fuzzy set we must use type reduction. Many type reductions like centroid, height or modified weight was presented but center-of-sets is common one.

\[ Y_{COS}(x) = [y_1, y_2] = \sum_{i=1}^{M} f^i y^i \]

(7)

Whole procedure for computing \( y_1 \) and \( y_2 \) is described in [6]. Crisp value from can be finally computed as

\[ y(x) = \frac{y_1 + y_2}{2} \]

(8)

4 Simulation environment

For our experiments we used simulation environments. All simulations were realized in Matlab environment using TrueTime simulator.

4.1 Matlab

The Matlab high-performance language for technical computing integrates computation, visualization in programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. MATLAB features a family of add-on application-specific solutions called toolboxes. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems.

4.2 TrueTime

TrueTime is a Matlab/Simulink-based simulator for real-time control systems. TrueTime facilitates co-simulation of controller task execution in real-time kernels, network transmissions, and continuous plant dynamics. The TrueTime network block simulates medium access and packet transmission in a local area network. Six simple models of networks are supported: CSMA/CD (e.g. Ethernet), CSMA/AMP (e.g. CAN), Round Robin (e.g. Token Bus), FDMA, TDMA (e.g. TTP), and Switched Ethernet.

The usage of the wireless network block is similar to and works in the same way as the wired one. To also take the path-loss of the radio signals into account, it has \( x \) and \( y \) inputs to specify the true location of the nodes. Two network protocols are supported at this moment: IEEE 802.11b/g (WLAN) and IEEE 802.15.4 (ZigBee). [13]

4.3 Simulation schema

Our simulation schema consists of PID controller, process and wireless network blocks. We used PID controller and transfer function of the process according to Tipsuwan [7]. Mathematical description of process \( G_p(s) \) and controller \( G_c(s) \) are followed:

\[ G_p(s) = \frac{2029.826}{(s+26.29)(s+22.96)} \]

\[ G_c(s) = \frac{0.1701(s+0.378/0.1701)}{s} \]

(9)

(10)

For representation of our ZigBee network blocks from TrueTime were used. Basic properties of whole network can be set in TrueTime Wireless Network block. For simulation we set mainly Network type, number of nodes, loss probability but the others can be set too. Network nodes are represented by TrueTime Send and TrueTime Receive blocks that provide data exchange. Simulation schema is on the next figure.
Type-2 fuzzy controller was developed based on PID controller equivalent. Universe of the error \((-1.5, 1.5)\) and integral of error \((-0.15, 0.15)\) was covered by 7 type-1 membership functions. Mean of output membership functions was computed for all combinations of inputs as:

\[
\text{Mean}_{\text{OUT}} = 0.1701 \text{Mean}_E + 0.378 \text{Mean}_{IE} \quad (11)
\]

This steps leads to type-1 fuzzy controller equivalent of PID controller. This is also type-2 fuzzy controller equivalent with one mean value according to (3). We create interval of mean values around this value for better reaction to the uncertainties. On the next figure you can see that PID controller and type-2 fuzzy controller (with only one mean value) are equivalent. In our simulations we set delta about mean values as 0.65 for error and 0.065 for integral of error. Sigmas were set 0.5 and 0.05.

\[
\text{System with 0.001s sampling time}
\]

5 Simulation results

With simulation schema on figure 5 we did three experiments to compare PID controller and type-2 fuzzy controller performance. All three settings can influence network or control system properties.

5.1 Sampling period

In networked control system sensors usually sample process values in periodic times. Different sampling times are suitable for different dynamics of systems. Usually more samples lead to much better control of the system. This is not always productive for networked control systems. Many sampling values can overload network communication. This could lead to network delays that could affect stability of whole system. In our first simulation we changed sampling periods against simulation of the figure 5 from 0.005s to 0.001 and you can see results on the next figure.

\[
\text{System with 14 nodes}
\]

5.2 Node numbers

There would be no reason for networked control if we want to control one system over network. Multiple systems are interconnected through network nodes. Just like with sampling period more nodes can overload network communication, especially with some network protocols. We added multiple network nodes with different sampling periods. This caused that more than one node wanted to transmit data at the same time at some point of simulation.

\[
\text{System with packet dropout}
\]

5.3 Packet dropout

With long distance between nodes some information could lost. In wireless networks signal loss and disturbance can occur often. When packet is lost retransmission of data is possible when the sampling period is not too small. Many times last received sample is used again for computation. Next figure show simulation of packet dropout without retry limit.
6 Conclusion
Recently, the progress in network technology over the past decade is bringing an advancing trend to control system, where communication networks replace point-to-point cables. This paper has introduced the fundamental characteristics of NCSs and communication network ZigBee.

In Matlab environment using TrueTime tool we simulated control loop with PID and type-2 fuzzy controller. Our goal was comparing performance of designed control systems. We focused on three problems of NCSs that are sampling period, network overload and packet dropout.

With PID controller each of this three issues decrease quality of control. In this case system had oscillating character. With type-2 fuzzy controller system also decrease control performance but compared to PID controller we obtained less overshoot and faster settle time. It is because type-2 fuzzy controller works with our issues as uncertainties.

Acknowledgement
This work was supported by project under number Req-00048-001.

References: